OIPE JC18
MAR 0 8 2004
PATENT & TRADEMARK OFFICE

6/A
EWm
3-16-04

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:

Louzoun

Application No.: 10/082,434

Filed: February 22, 2002

For:
**SYSTEM AND METHOD FOR EFFICIENT SCHEDULING OF MEMORY**

Examiner: Truong, Bao Q..

Art Unit: 2187

RECEIVED
MAR 1 2 2004
Technology Center 2100

**AMENDMENT**

ASSISTANT COMMISSIONER FOR PATENTS
Washington, D.C. 20231

Dear Sir:

In response to the Office Action, dated November 3, 2003, please make the following amendments and consider the following remarks:

---

A1

1.(currently amended) A method comprising:

receiving a plurality of commands to access at least one of a plurality of memory banks of a memory; and

scheduling the plurality of commands based at least in part on a status information of at least one of the plurality of memory banks; and

arbitrating between the commands to determine priority of access to the memory bus based at least in part on the status information

2. (original) The method of claim 1 wherein the memory is a synchronous dynamic random access memory.

3. (original) The method of claim 1 wherein the status information based at least in part on an idle state of the plurality of memory banks with respect to a bank based queuing scheme.

4. (original) The method of claim 1 wherein the status information is either an idle state of the plurality of memory banks.

A1

5. (original) The method of claim 1 wherein the status information is based at least in part on a type of a most recent command forwarded to the memory device via a memory bus.

6. (original) The method of claim 1 wherein the plurality of commands are read and write commands.

7. (currently amended) A system comprising:

a processor; and

a logic, coupled to the processor and to at least one memory device with a plurality of memory banks, to receive commands to access the memory device

and to schedule and arbitrate the commands based at least in part on a status information of the plurality of memory banks.

A1

8. (original) The system of claim 7 wherein the commands are read and write commands.

9. (original) The system of claim 7 wherein the wherein the status information is based at least in part on a type of most recent command forwarded to the memory device via a memory bus.

10. (original) The system of claim 7 wherein the plurality of memory banks perform in parallel.

11. (original) The system of claim 7 wherein the logic is a network switch or a memory controller.

12. (original) The system of claim 7 wherein the status information is an idle state of the plurality of memory banks.

13. (original) The system of claim 7 wherein the memory is a synchronous dynamic random access memory.

14. (currently amended) An apparatus comprising:

a first logic, coupled to at least one memory device with a plurality of memory banks, to receive commands to access the memory device; and

a second logic, coupled to the first logic, to schedule and arbitrate the received commands based at least in part on a status information of the plurality of memory banks.

15. (original) The apparatus of claim 14 further comprising a third logic to forward the schedule of the received commands to the memory device via a memory bus.

16. (original) . The apparatus of claim 14 wherein the apparatus is either one of a network switch or memory controller.

17. (original) The apparatus of claim 14 wherein the memory device is a synchronous dynamic random access memory.

18. (original) The apparatus of claim 14 wherein the received commands are read and write commands.

19. (original) The apparatus of claim 14 wherein the status information is based at least in part on a type of most recent command forwarded to the memory device via the memory bus.

20. (original) The apparatus of claim 14 wherein the plurality of memory banks perform in parallel.

21. (original) The apparatus of claim 14 wherein the status information is an idle state of the plurality of memory banks.

22. (currently amended)A method comprising:

receiving a plurality of commands to access at least one of a plurality of memory banks of a memory coupled to a memory bus;

scheduling the plurality of commands based at least in part on a status information of at least one of the plurality of memory banks; and

arbitrating between the commands to determine priority of access to the memory bus based at least in part on the status information.

23. (original) The method of claim 22 wherein the memory is a synchronous dynamic random access memory.

24. (original) The method of claim 22 wherein the status information is an idle state of the plurality of memory banks.

25. (original) The method of claim 22 wherein the plurality of memory banks perform in parallel.

26. (original) The method of claim 22 wherein the status information is based at least in part on a type of a most recent command forwarded to the memory device via a memory bus.

27. (original) The method of claim 22 wherein the plurality of commands are read and write commands.

28. (currently amended) An article comprising:

a storage medium having stored thereon instructions, that, when executed by a computing platform, result in forwarding a plurality of commands to a memory device by:

receiving the plurality of commands to access at least one of a plurality of memory banks of the memory device; and

scheduling and arbitrating the plurality of commands based at least in part on a status information of at least one of the plurality of memory banks.

29. (original) The article of claim 28 wherein the memory is a synchronous dynamic random access memory.

30. (original) The article of claim 28 wherein the status information is an idle state of the plurality of memory banks.

31. (original) The article of claim 28 wherein the plurality of memory banks perform in parallel.

32. (original) The article of claim 28 wherein the status information is based at least in part on a type of a most recent command forwarded to the memory device via a memory bus.

33. (original) The article of claim 28 wherein the plurality of commands are read and write commands.